Artificial Neural Networks

Hands-On Project: Building an Artificial Neural Network

Due date: Friday, December 12.

Building an artificial neural network to estimate the overall quality of a football player.

This activity aims to follow the construction process of a deep artificial neural network for a classification problem using the Fifa19 dataset*. The goal is to estimate the overall quality of football players based on their skills as a classification task. The raw dataset contains 18,207 instances with 89 attributes related to football players from latest edition FIFA19.

After preparing and cleaning the dataset, we end up with 16,134 instances with 22 attributes regarding football player skills: Crossing, Finishing, Heading Accuracy, Short Passing, Volleys, Dribbling, Curve, Free Kick Accuracy, Long Passing, Ball Control, Reactions, Shot Power, Stamina, Strength, Long Shots, Aggression, Interceptions, Positioning, Vision, Penalties, Composure, and Marking. These attributes have been chosen because their values correlate greater than 0.3 or less than -0.3 with the overall score (the variable to predict), which has been quantile-based discretized into four classes or categories: *Poor* football players with overall scores in [46, 62], *Intermediate* for values in [63, 66], *Good* players in [67, 71], and *Excellent* players for overall values in [72, 94].

Students, in **groups of four people**, will perform the following tasks:

1. Prepare the environment for Python with Tensorflow and Keras. You can choose any development environment, although Anaconda-Jupyter® is encouraged.
2. Clean and prepare the dataset.
    - Download the raw data file *FootballPlayerRawDataset.csv* and the notebook *PreparingFootballPlayerDataset.ipynb* from the Moodle platform, section *Practical Assignment (Units 1 abd 2)*, Hands-on project Units 1 and 2: Building an artificial neural network.
    - Upload the raw data file and the notebook under your *anaconda_projects* folder in your Anaconda-Jupyter server installation.
    - Open the notebook and check the file paths INPUT_FILE_NAME, ATT_FILE_NAME, and ONE_HOT_ENCODED_CLASSES_FILE_NAME to point to the raw-data file location and the resulting files with the attributes (inputs) and the target outputs (the one-hot-encoded-classes), respectively.
    - Execute the notebook to clean and prepare the data and obtain the resulting files to feed the neural models. Pay attention to the actions performed to understand the dataset better. The result of this process should be two CSV files: FootballPlayerPreparedCleanAttributes.csv and FootballPlayerOneHotEncodedClasses.csv. The former contains the prepared and clean instances with the attributes (predictors). The latter includes the corresponding one-hot encoded target classes for the overall quality of football players: poor, intermediate, good, and excellent.
3. Construct a deep neural network. Write a notebook that first implements the data loading process of the two .csv files: attributes and classes. Then, split the dataset into three partitions: 80% for training, 10% for development testing, and 10% for final testing purposes. Finally, **follow the deep-neural network construction process** to find the neural architecture and other hyperparameters

that achieve the best performance in classification accuracy. Grid search and other random or automatic techniques are permitted only in the final steps of the construction process to fine-tune the hyperparameters; however, we do not encourage their use. Consider a Bayesian error of 10% (minimum error), i.e., the human error that soccer scouts make when predicting the quality of football players from their skills. You can use the notebook that implements the deep neural model in Keras to estimate the median house value studied in class as a starting point for this task.

4. Write a report in Spanish or English describing the actions performed and the results achieved during this activity. The notebook developed in the previous task may be helpful in this regard. The structure of this report is described below. **The correctness of the construction process followed is essential**. It is also necessary to adequately employ the training, development, and final testing datasets at the right time.
5. **Send the report as a single PDF file** via Moodle by December 12. Make a single upload for all group members and keep the source code (notebook) in case the instructors require it during the revision process.

The structure of the report to write is the following:

1. Cover page. Include a cover page with title, authors, email, course, and date.
2. Introduction. Explain the problem to solve, the dataset, and the conclusions gathered from the statistical results shown after running *PreparingFootballPlayerDataset.ipynb*.
3. Design process. Describe the process you followed to get the final results, showing the intermediate network architectures and the rest of the hyperparameters used. Explain your design decisions, justifying why you tested each new neural model. Show the performance of each intermediate model.
4. Final results. Describe the ultimate neural network solution, clearly showing all the hyperparameters used. Display how the accuracy changes during the training process of this model. Calculate and analyze the confusion matrix for the final test set.
5. Conclusions. Summarize your work and the most relevant results.

Note: Exceptional cases of three- or five-person group sizes may be accepted. If so, please email martin.molina@upm.es and daniel.manrique@upm.es indicating the reasons for permission.

* https://www.kaggle.com/datasets/javagarm/fifa-19-complete-player-dataset.